**Application and Solution of the Heat Equation
in One- and Two-Dimensional Systems
Using Numerical Methods**

Computer Project Number Two

**By
Dr. David Keffer**

Table of Contents

List of Figures

I. Problem Description

Our study of heat transfer begins with an energy balance and Fourier's law of heat conduction. The first working equation we derive is a partial differential equation. The problem is that most of us have not had any instruction in how to deal with partial differential equations (PDEs). It becomes difficult to get a feel for heat transfer when we lack the mathematical tools to tackle even the most basic PDE. We are left the unsatisfying alternative of looking at plots of solutions and having them explained to us.

This situation has summed up the status quo with respect to the relationship between undergraduate chemical engineering education and heat transfer for as long as there has been undergraduate chemical engineering education. At least, that is how it appears to me. As undergraduate chemical engineers we usually don't take a PDE class and most of us never will. (Moreover, if the truth be told, when we do take a couple of PDE courses from a mathematics department, we find the solutions so onerous that we realize that we never want to solve PDEs analytically.) Our study of heat transfer appears to be sabotaged.

Historically, chemical engineers have gotten around this problem in several ways. The worst way is that chemical engineers ignore the problem and don't ever get a feel for heat transfer. A better (but not great) alternative is to look at a variety of limited one-dimensional cases with analytical solutions. Sometimes, we refer to a text of PDE solutions, as is to be found in a book like "Conduction of Heat in Solids" by Carslaw and Jaeger.[1] Even this approach is not very satisfying (because, again, the solutions accessible to us depend on their having been solved by Carslaw and Jaeger) and not so helpful (because even Carslaw and Jaeger is difficult to wade through when we are not familiar with PDEs).

Today, we are at a lucky juncture. We have available to us computational tools that make the solution of one- and two-dimensional PDEs a piece of cake. Decades ago, no undergraduate chemical engineer knew how to solve complex nonlinear ordinary differential equations (ODEs). Today, all undergraduate chemical engineers know at least the Runge-Kutta method for solving an ODE. We are at the beginning of the same transition for PDEs. All it takes is a few hours to sit down at a computer and see how the PDE software works. We can look at an unlimited selection of heat-transfer problems because we are not confined to problems with analytical solutions or problems that have been solved by Carslaw and Jaeger. Moreover, we have the opportunity to get a real feel for heat transfer by creating plots and movies of heat transfer in various situations. We should not be afraid to tackle PDEs with a tool like MATLAB. On the contrary, we should be delighted that we have the ability to study heat transfer problems in a way that suits the engineer. This way includes a few steps:

1. Describe the physical system
2. Translate the physical system into a PDE (an energy balance in the case of heat transfer), boundary conditions, and initial conditions
3. Enter the PDE, boundary conditions, and initial conditions into the software
4. Run the software
5. Analyze the solution

Now some of you may be thinking that you would have been happier in the dark ages of chemical engineering when undergraduates really had no conception of how to deal with PDEs. You may not be thinking kindly of Dr. Keffer for your forced exposure to PDEs, for which most of you probably hold no great fondness. Well, be patient. At the end of this project, you may have an appreciation for PDEs and a better grip on heat transfer. Keep an open mind until then. With this project, we intend to demonstrate to ourselves that we have the knowledge and capability to formulate and solve partial differential equations in one- and two-dimensional engineering systems. We will study the heat equation, a mathematical statement derived from a differential energy balance. We will describe heat transfer systems in terms of energy balances. We will enter that PDE and the appropriate boundary and initial conditions into MATLAB. With a minimum of effort, a vast territory of heat transfer problems, previous uninviting to the undergraduate chemical engineer, will be open to us.

II. Project Objectives

The objectives of this project are to

(1) Use computational tools to solve partial differential equations.
(2) Demonstrate the ability to translate a physical heat transfer situation into a partial differential equation, a set of boundary conditions, and an initial condition.
(3) Demonstrate the ability to formulate the PDE, the initial conditions, and boundary conditions in terms the software understands.
(4) Use existing MATLAB routines to solve
      (A) Steady-state One-dimensional heat transfer in a slab
      (B) Steady-state Two-dimensional heat transfer in a slab.
      (C) Unsteady-state One-dimensional heat transfer in a slab
      (D) Unsteady-state Two-dimensional heat transfer in a slab.
(5) Make quantitative statements about the physical meaning of the solutions of the PDEs, as they relate to engineering process variables of the system.
(6) Describe the physical behavior of the heat transfer system in terms of the features in the temperature profiles generated by the software
(7) Select process variables so as to achieve a desired heat transfer.

III. Report Requirements

The project requires a typed report. A collection of papers including MATLAB routines, plots, and numbers stapled together is unacceptable.

The report should follow the "Writing Guidelines for ChE 310 & 410" as composed by Dr. Weber and Dr. Bruns. If you do not have a copy of the handout, it is posted on the web at http://clausius.engr.utk.edu/che310/index.html . That procedure should be followed as closely as possible. Where the word "experimental" occurs in the report, substitute "numerical", such as "experimental methods" becomes "numerical methods" for this project.

Some features of the "Writing Guidelines for ChE 310 & 410" will not apply to this project. In that case, use your common sense and omit sections where necessary.

Briefly, the following components from the "Writing Guidelines for ChE 310 & 410" are required:

1. Title Page
2. Table of Contents, List of Figures, and List of Tables
3. Summary
4. Introduction and Background
5. Experimental Methods or Numerical Methods
6. Results and Discussion
7. Conclusions
8. Nomenclature (if at all different from the nomenclature used in this handout)
9. Literature cited (if references beyond this handout were used)
10. Appendices (This is where MATLAB routines would go.)
11. Figures

You do not need to attach a copy of this handout to the report. The report need not be wordy. Short and sweet is best. You do not need to repeat the derivations given in this handout.

When in doubt about how to proceed with some aspect of the report, rely on two resources: (1) "Writing Guidelines for ChE 310 & 410" as composed by Dr. Weber and Dr. Bruns, and (2) your common sense. Realize that this project description is another example of a report which follows the writing guidelines.

IV. Problem Formulation

In the analysis of a heat transfer system, as in all engineering systems, our first step should be to write out the appropriate balance equations. The generalized balance equation looks like this:

$$accum = in - out + gen - con \tag{1}$$

For heat transfer, our balance equation is one of energy. We will make several assumptions in formulating our energy balance.

- assumption 1. No momentum transfer. There is no heat transfer due to flow (convection) or due to a momentum gradient perpendicular to flow.
- assumption 2. No mass transfer. There is no heat transfer due to diffusion (due either to a concentration or thermal gradient).
- assumption 3. No shaft work. There is no energy input or output due to mechanical work.
- assumption 4. We have an incompressible material. The fluid density, ρ, is a function of neither pressure nor temperature.
- assumption 5. The enthalpy of the material can be approximated with an ideal entropy given by

$$\hat{H}(T) = \hat{C}_p(T - T_{ref}) \tag{2}$$

   where $\hat{H}$ is the enthalpy per unit mass and is a function only of temperature, $\hat{C}_p$ is the heat capacity per unit mass and is a constant, T_{ref} is the thermodynamic reference temperature.

- assumption 6. The heat capacity, $\hat{C}_p$, and the thermal conductivity, k, are not functions of temperature.
- assumption 7. The material is isotropic. The material properties, $\hat{C}_p$, k, and ρ, are the same at all points within the material. The thermal conductivity is the same in the x, y, and z directions.

Our system is an infinitesimal cube. It has a volume $V = \Delta x \Delta y \Delta z$. The accumulation of energy in the system is

$$acc = \frac{\partial\left(V\rho\hat{C}_p T\right)}{\partial t} = \Delta x \Delta y \Delta z \rho \hat{C}_p \frac{\partial(T)}{\partial t} [=] \frac{energy}{time} \tag{3.a}$$

We can pull the density, heat capacity, and volume out of the timer derivative because they are constants, as stated by our assumptions above. The in term in our energy balance is

$$in_x = \left(A_x q_x\right)_x = \left(\Delta y \Delta z q_x\right)_x [=] \frac{energy}{time}$$

$$in_y = \left(A_y q_y\right)_y = \left(\Delta x \Delta z q_y\right)_y [=] \frac{energy}{time}$$

$$in_z = \left(A_z q_z\right)_z = \left(\Delta x \Delta y q_z\right)_z [=] \frac{energy}{time}$$

The out terms in our energy balance are:

$$out_x = (A_x q_x)_{x+\Delta x} = (\Delta y \Delta z q_x)_{x+\Delta x} [=] \frac{energy}{time}$$

$$out_y = (A_y q_y)_{y+\Delta y} = (\Delta x \Delta z q_y)_{y+\Delta y} [=] \frac{energy}{time}$$

$$out_z = (A_z q_z)_{z+\Delta z} = (\Delta x \Delta y q_z)_{z+\Delta z} [=] \frac{energy}{time}$$

We will leave the generation term in our energy balance as a general function which has units of energy/time/volume. The function has a positive sign for generation and a negative sign for consumption.

$$gen = Vf(t,x,y,z) = \Delta x \Delta y \Delta z f(t,x,y,z) [=] \frac{energy}{time}$$

With these terms, we arrive at an energy balance that looks like:

$$\Delta x \Delta y \Delta z \rho \hat{C}_p \frac{\partial T}{\partial t} = (\Delta y \Delta z q_x)_x + (\Delta x \Delta z q_y)_y + (\Delta x \Delta y q_z)_z$$
$$- (\Delta y \Delta z q_x)_{x+\Delta x} - (\Delta x \Delta z q_y)_{y+\Delta y} - (\Delta x \Delta y q_z)_{z+\Delta z} + \Delta x \Delta y \Delta z f(t,x,y,z)$$

Divide by the volume. Reorder terms.

$$\rho \hat{C}_p \frac{\partial T}{\partial t} = \frac{(q_x)_x - (q_x)_{x+\Delta x}}{\Delta x} + \frac{(q_y)_y - (q_y)_{y+\Delta y}}{\Delta y} + \frac{(q_z)_z - (q_z)_{z+\Delta z}}{\Delta z}$$
$$+ f(t,x,y,z)$$

Take the limit as the infinitesimal elements approach zero. Recognize the definition of the derivative. Arrive at the energy balance:

$$\rho \hat{C}_p \frac{\partial T}{\partial t} = -\left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \right) + f(t,x,y,z) \tag{3}$$

where our energy balance has units of $\left[\dfrac{energy}{volume \cdot time} \right]$, where q is an energy flux with units of

$\left[\dfrac{energy}{area \cdot time} \right]$, and where $f(t,x,y,z)$ is a generalized function which incorporates all generation and consumption terms.

We must insert Fourier's law for heat conduction, which in the z-direction is:

$$q_z = -\alpha \frac{d(\rho C_p T)}{dz} \tag{4}$$

to obtain

$$\rho \hat{C}_p \frac{\partial T}{\partial t} = \alpha \rho \hat{C}_p \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + f(t, x, y, z) \tag{5}$$

We must realize that our thermal conductivity is defined as

$$k = \alpha \rho \hat{C}_p \tag{6}$$

so that the three-dimensional heat equation takes the form:

$$\rho \hat{C}_p \frac{\partial T}{\partial t} = k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + f(t, x, y, z) \tag{7}$$

or, simplifying:

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + \frac{f(t, x, y, z)}{\rho \hat{C}_p} \tag{8.a}$$

which in two-dimensions becomes

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + \frac{f(t, x, y, z)}{\rho \hat{C}_p} \tag{8.b}$$

and which in one-dimension becomes

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} \right) + \frac{f(t, x, y, z)}{\rho \hat{C}_p} \tag{8.c}$$

where , α , the thermal diffusivity is defined by rearranging equation (6)

$$\alpha = \frac{k}{\rho \hat{C}_p} \tag{9}$$

and has units of $\left[\dfrac{\text{length}^2}{\text{time}} \right]$, just like the diffusivity. The reason that we use α , instead of $\hat{C}_p$, k , and ρ ,

is that our problem reduces to one parameter, rather than three parameters. Of course, to solve the problem, we must first go to the literature and look up $\hat{C}_p$, k , and ρ , then calculate α .

It should be noted that equations (8.a-c) are respectively the one-dimensional, two-dimensional, and three-dimensional heat equations. They are first-order partial differential equations with respect to time. They are second order partial differential equations with respect to the spatial coordinates.

The physical meaning of the heat equation is the same physical meaning as that of an energy balance. It is an unsteady-state energy balance as we have written it.

In the one-dimensional case, the energy balance will be integrated over some system where the temperature varies only in the x-direction. One can imagine a case where we have a solid cylindrical beam connecting two heat reservoirs at different temperatures. (See Figure 1(a).) In this case, we neglect the radial and angular dimensions of the cylinder and say that the temperature is only a function of x (and of t, if we are not at steady state). We could also approximate a slab connecting these two reservoirs, where we neglect the width and depth of the slab, focussing only on its length. (See Figure 1(d).)

In the two-dimensional case, the energy balance will be integrated over a system where the temperature varies in the x-direction and the y-direction. One can imagine a case where we have a thin plate. The shape of the plate can change but it is still subject to the heat equation at any point with in the plate. (See Figure 2(a-b).)
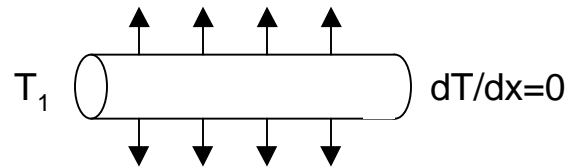
cylindrical rod
neglect radial and angular dimensionality

T_1 ◯⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍ T_2
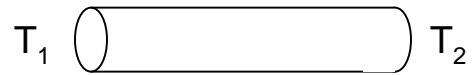
(a) 2 Dirichlet boundary conditions

T_1 ◯⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍ $dT/dx=0$

(b) 1 Dirichlet boundary condition,
      1 Neumann boundary condition

T_1 ◯⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍⟍ $dT/dx=0$

(c) 1 Dirichlet boundary condition,
      1 Neumann boundary condition
      heat loss to surroundings

rectangular slab
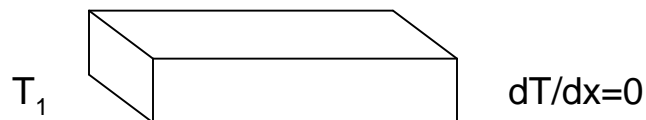**neglect width and depth dimensionality**

T_1 ▭ T_2
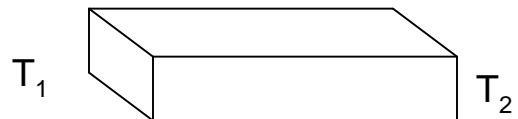
(d) 2 Dirichlet boundary conditions

T_1 ▭ $dT/dx=0$

(e) 1 Dirichlet boundary condition,
      1 Neumann boundary condition

Figure One. One-dimensional heat transfer systems

rectangular slab
neglect depth dimensionality

T_4 at y=L

T_2 at x=L

T_1 at x=0

T_3 at y=0

(a) 4 Dirichlet boundary conditions

slab of arbitrary geometry
(e.g. disk with a cut-out square)
neglect depth dimensionality

dT/dx =1

dT/dy =0

dT/dy =0

T_1
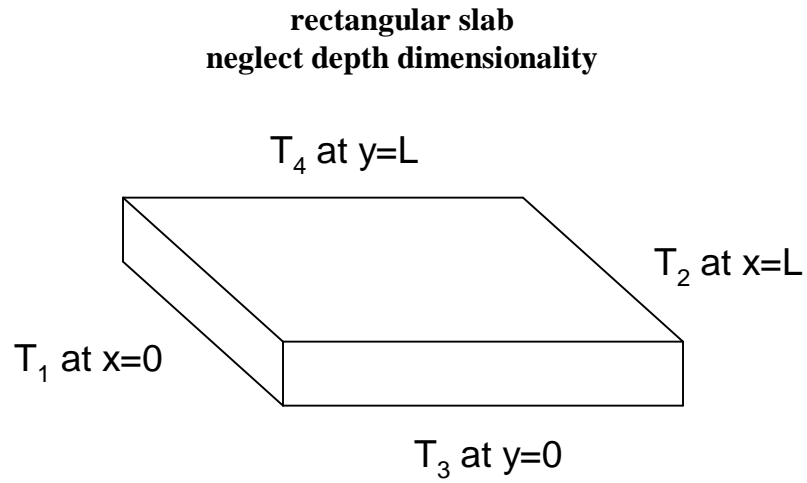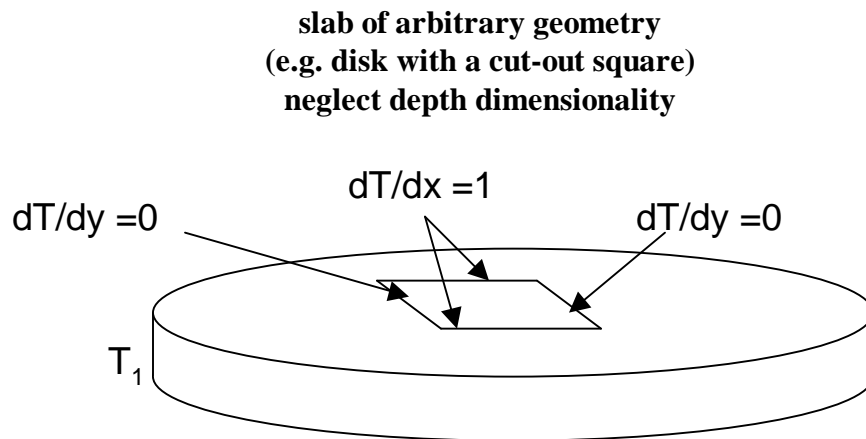
(b) Dirichlet boundary condition on exterior boundary
Neumann boundary condition on interior boundary

Figure Two. Two-dimensional heat transfer systems

V. Boundary Conditions

The heat equation is a second-order partial differential equation in the spatial coordinates. We need boundary conditions in order to specify how our system interacts with the outside surroundings.

There are three general types of boundary conditions: Dirichlet, Neumann, and Mixed boundary conditions.

1. Dirichlet Boundary Conditions

Dirichlet (pronounced Deer-reach-lay, with the accent on reach) boundary conditions say that the temperature is set at the boundary. They have a form like this (for the one-dimensional case)

$$T(x = 0, t) = T_{bc1}(t) \tag{10}$$

This says that at the left-hand-side boundary of our one-dimensional system, the temperature is a specified function of time. If the temperature is constant, then we have

$$T(x = 0, t) = T_{bc1} \tag{11}$$

In this case, we have the physical situation where our system is touching an infinite heat reservoir that maintains a constant temperature.

In a one-dimensional system, we must have two boundary conditions, one at the left-hand-side boundary and the other at the right-hand-side boundary. If our one-dimensional system is of length L in the x-direction, then our second Dirichlet boundary condition would be of the form:

$$T(x = L, t) = T_{bc2}(t) \tag{12}$$

2. Neumann Boundary Conditions

Neumann (pronounced noy-men, with the accent on noy) boundary conditions say that the heat flux is set at the boundary. They have a form like this (for the one-dimensional case)

$$\frac{dT}{dx}(x = 0, t) = \frac{dT(t)}{dx}\bigg|_{bc1} \tag{13}$$

This says that at the left-hand-side boundary of our one-dimensional system, the heat flux is a specified function of time. If the heat flux is constant we have

$$\frac{dT}{dx}(x = 0, t) = \frac{dT}{dx}\bigg|_{bc1} \tag{14}$$

In this case, we have the physical situation where our system is touching an infinite heat source that maintains a constant flux of heat into the system *regardless of the temperature*. When might this situation occur? One end of the rod is well insulated. No heat leaves it. The heat flux is zero. In this case, we would use a Neumann boundary condition.

In a one-dimensional system, we must have two boundary conditions, one at the left-hand-side boundary and the other at the right-hand-side boundary. If our one-dimensional system is of length L in the x-direction, then our second Neumann boundary condition would be of the form:

$$\frac{dT}{dx}(x=L,t) = \frac{dT(t)}{dx}\bigg|_{bc2} \tag{15}$$

3. Mixed Boundary Conditions

Mixed Boundary Conditions, as the name implies, is a mixture of the Dirichlet and Neumann boundary conditions. They have a form like this (for the one-dimensional case)

$$\frac{dT}{dx}(x=0,t) + T(x=0,t) = T_{bc1}(t) + \frac{dT(t)}{dx}\bigg|_{bc1} = f(t) \tag{16}$$

There are very relevant physical systems which require these elaborate boundary conditions.

4. Number of Boundary Conditions required

As state above, you must have two boundary conditions for the one-dimensional heat equation. There is no reason you can't have a physical system where you have a Dirichlet boundary condition at one end of your system and a Mixed boundary condition at the other. For example, consider the system where you have one end of a cylindrical rod in contact with a heat reservoir at a constant temperature (a Dirichlet boundary condition) but the other end exposed to the atmosphere, where heat loss occurs to the surroundings. The heat loss can be modeled with a heat transfer coefficient so that your second boundary condition becomes a statement of heat transfer:

$$qA = hA\left(T_{surround} - T(x,t)\right) \tag{17}$$

Fick's law says:

$$q_x = -\alpha \frac{d(\rho C_p T)}{dx}$$

Our heat loss at the boundary is then written as

$$-\alpha \frac{d(\rho C_p T)}{dx} A = hA\left(T_{surround} - T(x,t)\right)$$

so that

$$\frac{dT}{dx} = -\frac{h\left(T_{surround} - T(x,t)\right)}{k} \tag{18}$$

so that our boundary condition becomes:

$$\frac{dT}{dx}(x=L,t) = -\frac{h\left(T_{surround} - T(x=L,t)\right)}{k} \tag{19}$$

which is a Mixed boundary condition because it contains both the temperature and the flux of the rod.

Note Mixed boundary conditions are sometimes referred to as "generalized Neumann boundary conditions". (This is how the MATLAB pdetool refers to them.)

5. Two-dimensional case

In two-dimensions, we again require boundary conditions along all boundaries. In the systems shown in Figure 2, we must define Dirichlet, Neumann, or mixed Boundary conditions for all boundaries. If we had constant-temperature Dirichlet boundary conditions for the rectangle in Figure 2(a), we would have four boundary conditions that looked like

$$T(x = 0, y, t) = T_{bc1}, \ T(x = L, y, t) = T_{bc2}$$
$$T(x, y = 0, t) = T_{bc3}, \ T(x, y = L, t) = T_{bc4} \tag{20}$$

VI. Initial Conditions

1. Generalized initial condition

The heat equation is first order in time. We need to know the temperature at every point in our system at time equals zero. In general, this initial condition looks like:

$$T(x, y, t = 0) = T_{ic}(x, y) \tag{21}$$

2. Constant temperature initial condition

If the temperature is constant then the initial condition becomes:

$$T(x, y, t = 0) = T_{ic} \tag{22}$$

3. Steady-state initial condition

If the temperature profile is initially a steady-state (linear) profile between two boundary condition temperatures T_{bc1} and T_{bc2}, then we would have the formula for the linear interpolation between them, which in one-dimension looks like:

$$T(x, t = 0) = T_{bc1} + \frac{x}{L}\left(T_{bc2} - T_{bc1}\right) \tag{23}$$

VII. Generation and Consumption Terms

1. Generalized generation and consumption terms
     The heat equation as we have written it in equation 8(a-c) contains a generic generation and consumption term, which can be a function of time and position, and which, in one dimension, appears in the heat equation as $f(t,x)$,

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial x^2}\right) + \frac{f(t,x)}{\rho \hat{C}_p} \tag{8.c}$$

2. No generation or consumption
          When there are no generation or consumption terms,

$$f(t,x) = 0 \tag{24}$$

3. Non-Uniform heat loss or heat gain
          When we have heat loss (or gain) along the rod due to lack of insulation, the energy loss (or gain) [energy/time] has the form

$$qA = hA\left(T_{surround} - T(x,t)\right) \tag{15}$$

Our consumption (or generation) term per unit volume of the material [energy/time/volume]is then

$$f(t,x) = \frac{qA}{V} = \frac{hA\left(T_{surround} - T(x,t)\right)}{V} \tag{25}$$

which, substituting equation (15) into equation (8.c) gives us a heat equation of the form:

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial x^2}\right) + \frac{hA\left(T_{surround} - T(x,t)\right)}{\rho \hat{C}_p V} \tag{26.a}$$

The two-dimensional version would look like:

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) + \frac{hA\left(T_{surround} - T(x,y,t)\right)}{\rho \hat{C}_p V} \tag{26.b}$$

The pdetool used for solving the two-dimensional heat equation asks for the input in the format:

$$u' - \text{div}(c(\text{grad}(u))) + au = f$$

We can rearrange equation (26.b) to give us the same form:

$$\frac{\partial T}{\partial t} - \alpha\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) + \frac{hA\ T(x,y,t)}{\rho \hat{C}_p V} = \frac{hA\ T_{surround}}{\rho \hat{C}_p V}$$

We must make the following equalities:

$$u = T, \qquad u' = \frac{\partial T}{\partial t}, \qquad div((grad(u)) = \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

$$d = 1, \qquad c = \alpha, \qquad a = \frac{hA}{\rho \hat{C}_p V}, \qquad f = \frac{hA\ T_{surround}}{\rho \hat{C}_p V}$$

We know everything in these equations but the area and the volume.

For the two-dimensional case, the area is the surface area of the plate, exposed to the surrounding temperature. For a rectangular plate of length L_x and width L_y with one heat loss through one face, $A = L_x L_y$. For the same dimensions with heat loss through both faces, $A = 2L_x L_y$. The heat loss through the x and y boundaries is taken care of by the boundary conditions, not the generation term so that surface area does not appear in these equations. The volume would be $V = L_x L_y L_z$, where L_z is the width of the plate. This width appears no where else in the P.D.E. but is needed for this type of generation term.

For the case of a one-dimensional cylindrical rod, the area is the surface area of the rod, $A = \pi D L$. This diameter appears no where else in the P.D.E. but is needed for this type of generation term. The volume then is $V = \dfrac{\pi D^2 L}{4}$.

VIII. Solution Strategy

Once we have the properly posed problem, which contains the partial differential equation with the correct generation term, the boundary conditions, and the initial condition, we are ready to solve the problem. What is the solution of a P.D.E.? A solution of a P.D.E. is the value of the function, T, at every point in space at every instant in time. The solution to the heat equation is thus the temperature profile at every instant in time. Of course, we approximate the solution and find the temperature profile on a discretized grid (rather than at every point in space) and at distinct time intervals. We will use some numerical methods to solve this problem.

1. Solution strategy for the one-dimensional heat equation

We will use the Crank-Nicolson method for solving the one-dimensional parabolic PDEs.[2] (The heat equation is a member of a family of PDEs designated as parabolic.) The Crank-Nicolson method is an example of a "finite-difference" scheme, a technique used to solve one-dimensional PDEs. In this method, we break down the length in the x-direction into n intervals. We will then approximate the real continuous temperature profile with a discrete temperature profile. (See Figure Three.) We can solve this temperature profile again and again so that we obtain the temperature profile as a function of time. (See Figure Four). If we want to solve for the profile at time t_f, starting from t_o, then we must break that time period into m equal intervals. So now we have n spatial intervals and m temporal intervals. If we increase n and m, then we will achieve a more accurate solution. However, our computation time will increase. We have to find a balance between getting a reasonable result within a reasonable amount of time.

This solution strategy is implemented in the MATLAB code heateqn1d.m. Additional information on heateqn1d.m is given in Appendix II.

2. Solution strategy for the two-dimensional heat equation

We will use a finite element method for solving the one-dimensional parabolic PDEs.[2] The finite element method allows us to tackle PDEs in systems with unusual geometries, problems that go beyond rectangles and circles. Finite element techniques get ugly fast. However, MATLAB has provided a tool called "pdetool" that allows us to solve any two-dimensional heat equation using finite element methods. Having never used "pdetool" before, I was able to sit down with the User's manual[4] and master "pdetool" in two to three hours. I expect that the students can do the same. What is required here is the ability to sit down with a new piece of software and a tutorial and figure out, in a hands-on way, how it works. I think we all can agree that this is a useful skill, given how much new software is coming out and how quickly a piece of software can be dated. There is a copy of the User's Guide, "MATLAB Partial Differential Equation Toolbox" in room 314. Please do not take it from Room 314.

I will only summarize the procedure to use pdetool. At the MATLAB command prompt type "pdetool".

1. Draw your system geometry
2. Define your PDE (remember: the heat equation is a parabolic PDE)
3. Define your Boundary conditions
4. Define your initial conditions (in the "parameter" option of the "Solve" menu)
5. Define how long you want to model the system (in the "parameter" option of the "Solve" menu)
6. Select your plot options and plot the solution

It's very straight-forward. You should be able to have some fun with this tool. You can make movies and see how a particular heat transfer system reaches steady-state. Here are a couple hints:

- You must select a time-scale (step 5 above) which shows the evolution of the system. If the time scale is too small, you won't see much heat transfer. If the time-scale is too large, your first step will take you to steady-state.
- If the movies come out a gray blur, switch your monitor from True Color to 65K 16-bit color. This is a MATLAB bug.
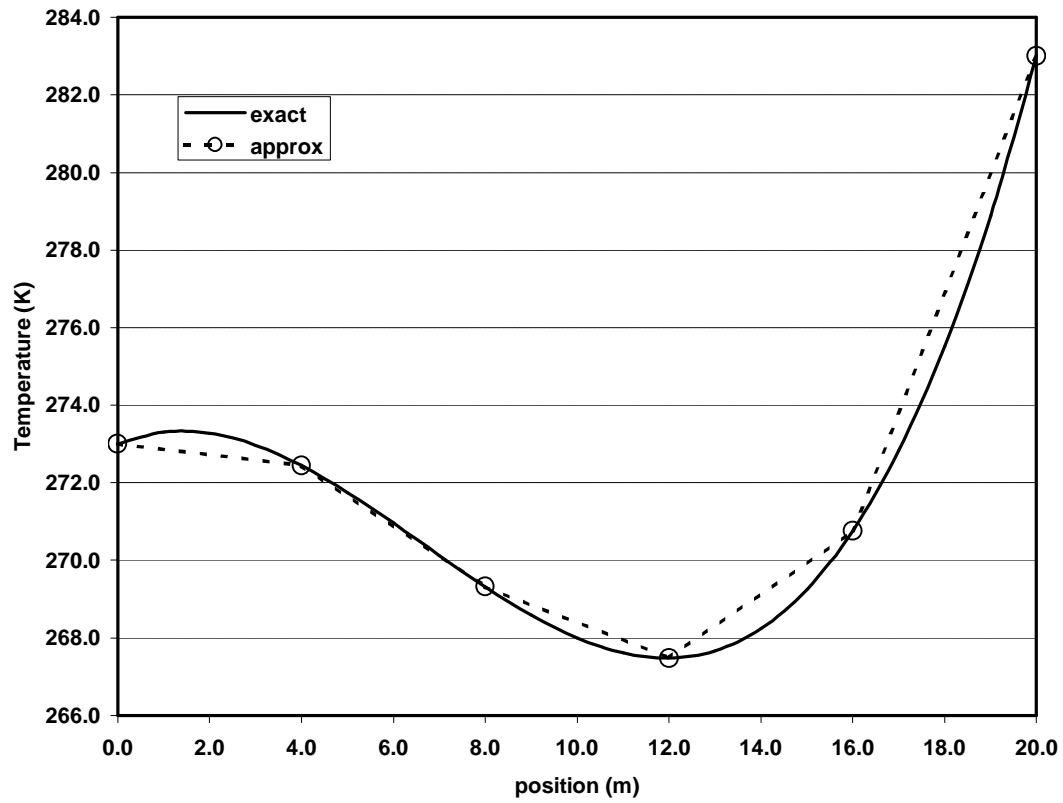
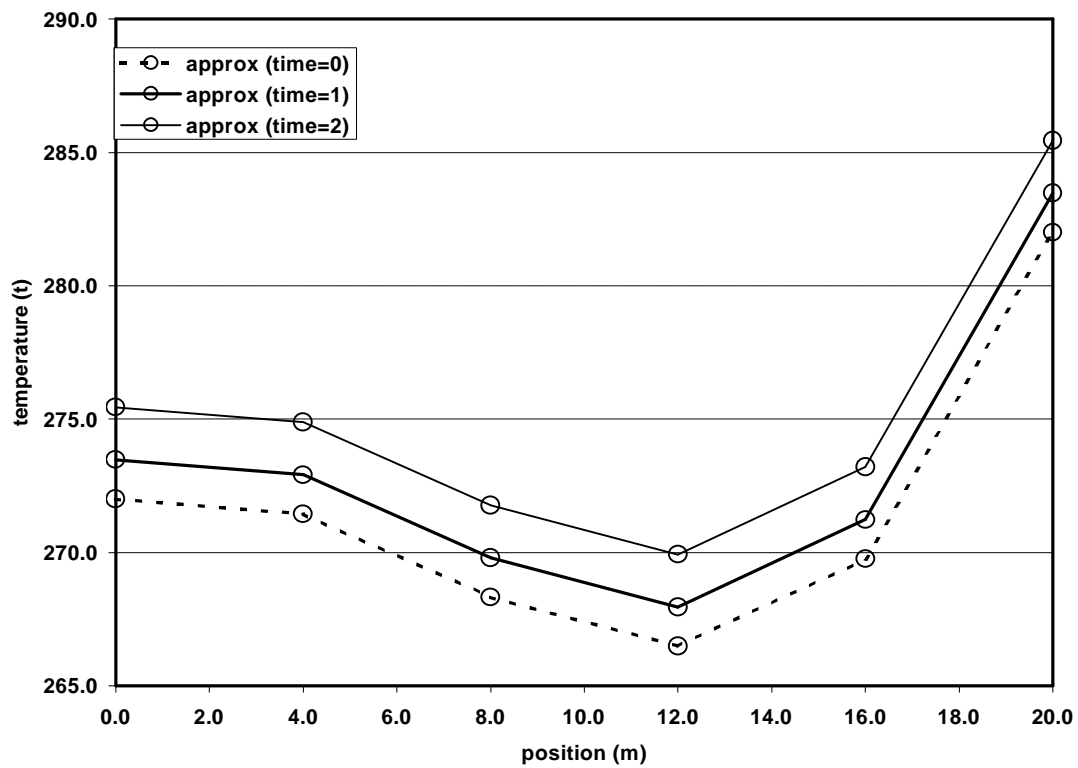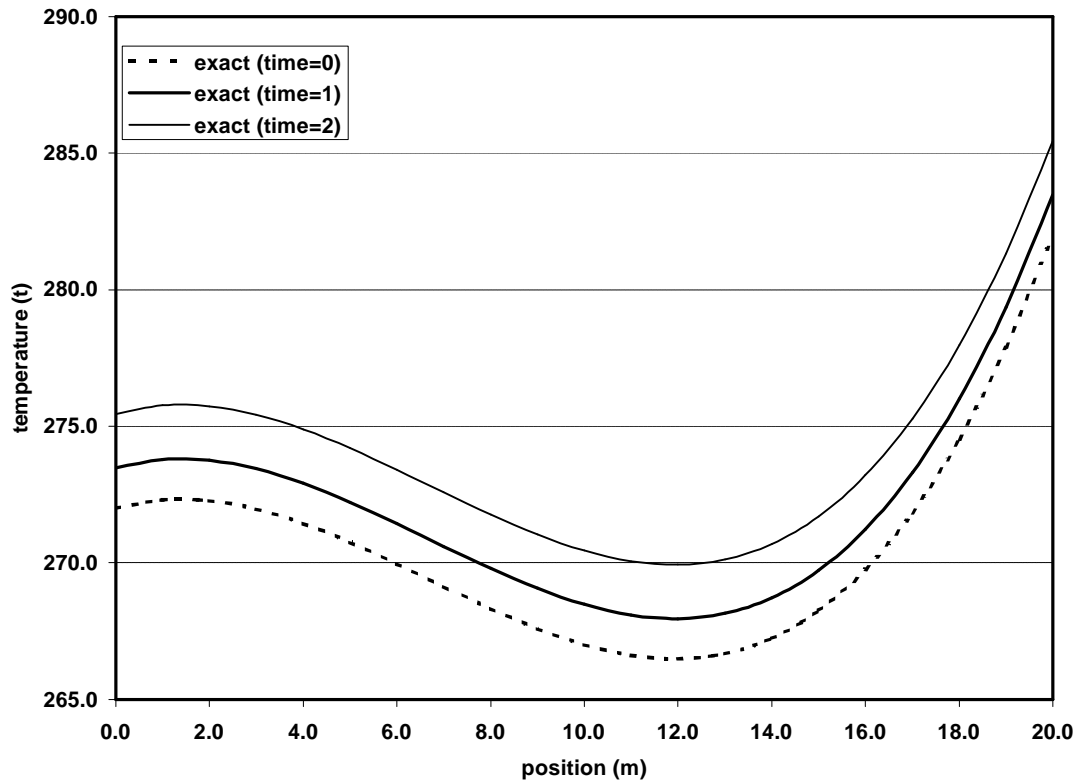Figure Three. Discretized spatial approximation to a temperature profile

Figure Four. Discretized temporal approximation to an evolving temperature profile. Exact solution (top). Approximate solution (bottom).

IX. MATLAB Requirements

Students are required to use MATLAB routines to solve the system of algebraic equations arising from the piping network analysis. Students have two options. Either option is perfectly acceptable and neither option is favored by the instructor.

Option 1. Write a MATLAB code from scratch.

- Advantage: Some students feel that they want to write their codes themselves in order to cement their understanding of the project.

- Disadvantage: Major time commitment to programming and debugging a piping network code. Additional time commitment if familiarity with MATLAB is sketchy.

Option 2. Modify and use a pre-existing MATLAB code.

- Advantage: The instructor has written a code which is capable of solving the problems specified in this report. This reduces the time commitment involved in programming and debugging the code. Modifying an existing code may be more feasible for students who are not confident in their MATLAB skills.

- Disadvantage: The student exchanges the time commitment of programming the code to reading and understanding the existing code. The instructor will provide adequate documentation to use the code "as is". Thus the instructor will not be available for extended tutorials on the use of the code. (This simulates life in research and industry, where as part of a group, you inherit someone else's code. Frequently, the original author of the code is nowhere to be found. You must be able to decipher the code yourself, with the help of documentation, if there is any, which frequently there is not)

- Disadvantage: There is the temptation, when using a pre-existing code, to try to use the code without opening and studying the code, and thus, without understanding how the code works. Typically, this results in totally useless results from the computer. Be warned: just because the instructor makes this code available does not absolve the student of the responsibility to understand how the code functions and to modify it where necessary.

For the two-dimensional case, students must use the MATLAB pdetool.

X. Project Requirements

(1) Consider an aluminum cylindrical rod 1.0 meter long connecting two heat reservoirs. One of the reservoirs is maintained at T=300K, the other reservoir at T=400 K. Initially, the cylinder is at 300 K. There is no heat loss from the rod. Consider the system to be one-dimensional.

      (a) Write the PDE, IC, and BC's.
      (b) What does the initial profile look like?
      (c) What does the steady state profile look like? Explain.
      (d) What is the temperature 0.5 meters into the rod at steady state?
      (e) What is the temperature 0.5 meters into the rod after 1000 seconds?
      (f) Approximately how long does it take for the midpoint of the rod to get within 1% of the steady state value?
      (g) Approximately how long does it take for the midpoint of a lead (Pb) rod to get within 1% of the steady state value? Explain the difference.

(2) Consider an aluminum cylindrical rod 1.0 meter long with one end connected to a heat reservoir at T=400 K. The other end is insulated.   The entire rod is also insulated so that there is no heat loss to the surroundings. The initial temperature of the rod is 300 K.

      (a) Write the PDE, IC, and BC's.
      (b) What does the steady state profile look like? Explain.
      (d) What is the temperature at the free end of the rod at steady state?
      (e) What is the temperature 0.5 meters into the rod after 1000 seconds?
      (f) Approximately how long does it take for the end of the rod to get within 1% of the steady state value?
      (g) Approximately how long does it take for the end of a gold (Au) rod to get within 1% of the steady state value?

(3) Consider an aluminum cylindrical rod 1.0 meter long connecting two heat reservoirs. One of the reservoirs is maintained at T=300K, the other reservoir at T=400 K. Initially, the cylinder is at 300 K. The rod is not insulated so heat is lost from the rod, which has a radius of 10 cm. The surrounding temperature is 200 K. Use a heat transfer coefficient of4.0 kW/m^2/K. Consider the system to be one-dimensional.

      (a) Write the PDE, IC, and BC's.
      (b) What does the steady state profile look like? Explain.
      (c) What is the temperature 0.5 meters into the rod at steady state?
      (d) What is the temperature 0.5 meters into the rod after 1000 seconds?
      (e) Approximately how long does it take for the midpoint of the rod to get within 1% of the steady state value?
      (f) Approximately how long does it take for the midpoint of a copper (Cu) rod to get within 1% of the steady state value? Explain the difference.
      (g) Demonstrate for me the concept of a wire's "critical thickness", as described by Geankoplis on page 231-232, by showing me the steady-state temperature profiles for aluminum cylinders of three different radii, one at the critical radius, one well below the critical radius, and one well above the critical radius.

(4) Consider an aluminum cylindrical rod 1.0 meter long connecting two heat reservoirs. Both of the reservoirs are maintained at T=300K. Initially, the cylinder is at 300 K, except for the center point of the cylinder which has been rapidly irradiated to a temperature of 600K. There is no heat loss from the rod. Consider the system to be one-dimensional.

      (a) Write the PDE, IC, and BC's.
      (b) What does the initial profile look like?
      (c) What does the steady state profile look like? Explain.
      (d) What happens to the energy pulse initially in the center of the rod?
      (e) What is the temperature 0.5 meters into the rod after 1000 seconds?

(5) Consider a two-dimensional rectangular aluminum plate. The boundary at x=0 is held at a constant temperature. The two y boundaries are insulated and have zero flux. The boundary at x=1 is held at a controlled, oscillating temperature.

$$T(x = 0, y, t) = 300K, \ T(x = 1, y, t) = 400K + 10\cos(t/10)$$
$$\frac{dT(x, y = 0, t)}{dy} = 0, \ \frac{dT(x, y = 1, t)}{dy} = 0$$

The initial condition of the slab is the steady state solution when the second x boundary is 400 K:
$$T(x, y, t = 0) = 300 + x \cdot (400 - 300)$$

There is no generation in the slab.

        (a) What does the initial profile look like?
        (b) What does the profile look like at 1000 seconds?
        (c) Describe the phenomena you observe from the movies of the temperature profile?

(6) Play around with the MATLAB routine pdetool. Create a system that models a heat transfer system from your experience.
        (a) Write down the PDE, the boundary conditions, and the initial conditions.
        (b) What does the initial profile look like?
        (c) If there is a steady-state, what does the steady-state temperature profile look like?
        (d) Show me one of the transient profiles.
        (e) Describe in words the behavior of the system.

Appendix I. Nomenclature

A surface area [m^2]

$\hat{C}_p$ heat capacity per unit mass [J/kg/K]

k thermal conductivity [W/m/K]

$\hat{H}$ enthalpy per unit mass [J/kg]

L length of one-dimensional system [m]

m number of temporal intervals [dimensionless]

n number of spatial intervals [dimensionless]

q heat flux [W/m^2]

t time [s]

T temperature [K]

T_{ref} thermodynamic reference temperature [K]

x spatial coordinate [m]

y spatial coordinate [m]

z spatial coordinate [m]

α thermal diffusivity [m^2/s]

ρ fluid density [kg/m^3]

Appendix II. Documentation for MATLAB code, "heateqn1d.m"

The code "heateqn1d.m" will solve the heat equation in one-dimension with arbitrary boundary conditions, initial conditions, and generation terms. The code is available on the course website. You must download it and unzip it using WinZip. Then you must move to whichever directory you have unzipped the files into. The zipped file contains the MATLAB file, "heateqn1d.m". The code is executed by typing at the MATLAB command line interface, for example.

» heateqn1d(4)

There is one input argument. For help on how to use the code, type:

» help heateqn1d

which displays:

```
The function heateqn1d solves a Parabolic Partial Diffential Equation
with one spatial dimension.  This is the unsteady-state one
dimensional heat equation.

accumulation - in/out       = generation
dT/dt        - alpha*d2T/dx2 = fgen(t,x)

This function handles space-dependent initial conditions,
time- and space-dependent Dirichlet and Neumann boundary conditions,
and time- and space-dependent heat generation/loss terms.

This function uses the Crank-Nicolson Method,
which is second order in both time and space derivatives,
as outlined in "Numerical Methods for Engineers" by
Steven C.Chapra and Raymond P. Canale, 2nd Ed., 1985,
McGraw-Hill, New York, p.742.

the function is executed at the command line by typeing
heateqn1d(lplot)
where lplot is an integer which selects the type of plot
(See comments in STEP ZERO of code for plot types.)

In order to alter this code to suit your personal needs,
You must check (at the top of the program):
(1) STEP ZERO.  Choose what kind of printed and plotted output you want.
(2) STEP ONE.       Choose the correct time and space intervals and final values
(3) STEP TWO.A.  Input physical properties (density, conductivity, heat capacity)
(4) STEP TWO.B.  Choose Neumann or Dirichlet boundary conditions
You must check at the bottom of the program):
(5) SET THE CORRECT INITIAL CONDITIONS
(6) SET THE CORRECT BOUNDARY CONDITIONS
(7) SET THE CORRECT HEAT GENERATION TERM
All the rest of the code need not be touched.

This code was written by David Keffer in the Department of Chemical Engineering
at the University of Tennessee, Knoxville, around about December 19, 1998.
```

Each step of the code is filled with comments. You'll need to read the comments to understand the code.

References

1. Carslaw, H.S., Jaeger, J.C., "Conduction of Heat in Solids", 2 nd Ed., 1959, Clarendon Press, Oxford.

2. Chapra, Steven C., Canale, Raymond P., "Numerical Methods for Engineers", 2nd Ed., 1985, McGraw-Hill, New York, p.742.

3. Geankoplis, Christie J., "Transport Processes and Unit Operations", Third Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1993, pg. 86-100.

4. "MATLAB Partial Differential Equations Toolbox User's Guide", The Mathworks Inc., Natick, MA, 1995.